Section 2: Installing CPExpert under MVS

CPExpert normally is distributed on a non-labeled tape containing two source format libraries: SOURCE and USOURCE. The tape was created using IBM's IEBUPDTE utility software. This section describes how to install CPExpert on a mainframe system operating under MVS.

Chapter 1: Unload the distribution tape, using IEBUPDTE.

The first step is to unload the distribution tape containing the CPExpert code. The PDS requires fifteen to fifty cylinders of IBM-3380 space (depending upon how many components of CPExpert you have ordered), and consists of members containing standard 80-byte record.

Use the Job Control Language (JCL) shown in Exhibit 2-1 to unload the data from tape to disk. This JCL unloads the majority of the CPExpert source code into CPExpert's normal source library (titled SOURCE), and unloads some CPExpert source code into a user source library (titled USOURCE). The USOURCE library contains parameters that specify general guidance to CPExpert (GENGUIDE), and specify guidance for each CPExpert component (xxxGUIDE, where "xxx" identifies the CPExpert component to which the guidance relates).

```
//jobname     JOB     job card information
//STEP01      EXEC    PGM=IEBUPDTE,PARM=NEW
//SYSPRINT    DD      SYSOUT=*
//SYSIN       DD      DSN=CPEXPERT.SOURCE,DISP=(OLD,KEEP),UNIT=TAPE9,
//                    VOL=(,RETAIN,SER=CPE000),LABEL=(1,NL,EXPDT=98000),
//                    DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//SYSUT2      DD      DSN=prefix.CPEXPERT.SOURCE,DISP=(,CATLG,DELETE),
//                    UNIT=3380,VOL=SER=volser,SPACE=(CYL,(15,10,40)),
//                    DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//STEP02      EXEC    PGM=IEBUPDTE,PARM=NEW
//SYSPRINT    DD      SYSOUT=*
//SYSIN       DD      DSN=CPEXPERT.USOURCE,DISP=(OLD,KEEP),UNIT=TAPE9,
//                    VOL=SER=CPE000,LABEL=(2,NL,EXPDT=98000),
//                    DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//SYSUT2      DD      DSN=prefix.CPEXPERT.USOURCE,DISP=(,CATLG,DELETE),
//                    UNIT=3380,VOL=SER=volser,SPACE=(TRK,(2,1,10)),
//                    DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
```

JOB CONTROL LANGUAGE TO INSTALL CPExpert UNDER MVS

EXHIBIT 2-1

The JCL is only a pattern. You will need to make the following changes:

JOBNAME	JOB	The job card information must be completed according to your installation's requirements
SYSIN	DD	The VOLSER CPEnnnn should be replaced by the actual VOLSER printed on the tape you receive.
SYSUT2	DD	The VOLSER=SER=volser must be replaced by the DASD volume on which you plan to unload the SOURCE. You may also need to change the unit type to SYSDA, based on your installation standards. Additionally, change the prefix in accordance with your installation standards.

Please use data set names (DSN) with USOURCE and SOURCE as the final qualifier to facilitate communications about the software. You will not normally make changes to any member in the SOURCE PDS. However, you **will** make changes to members in the USOURCE PDS. These changes are described in the appropriate User Manuals for CPExpert components.

The SYSUT2 specification for USOURCE needs special attention. This is because CPExpert periodically writes information to the USOURCE PDS. CPExpert normally[1] reports the guidance specified in USOURCE for each CPExpert component, after the component executes. This listing is so a user can relate the analysis that was performed with any unique (or "override") guidance given to the CPExpert component. Some CPExpert users have written code to dynamically specify guidance, so the report produced by CPExpert cannot be a simple listing of the lines in a USOURCE guidance member.

Consequently, CPExpert processes the USOURCE member, and dynamically creates a SAS macro, which is stored in USOURCE. CPExpert then executes a %INCLUDE of the SAS macro, and the macro lists the then-current value of each guidance variable. Since CPExpert writes the macro to USOURCE, the USOURCE library can become full. There are two ways to prevent this situation: (1) create USOURCE as a PDSE, or (2) periodically COMPRESS the USOURCE library.

The names of the members of the PDS generally denote the component with which they are associated. The component is identified by the first three characters of the PDS member (e.g., WLMxxxxx, DASxxxxx, CICxxxxx, etc.).

[1] This listing of guidance variables can be suppressed by specifying %LET LISTGDE=N in USOURCE(GENGUIDE) to suppress listings for all components, or specify %LET LISTGDE=N in USOURCE(xxxGUIDE) to suppress listing for a specific component identified by the "xxx" notation.

Members used by more than one component are prefixed with the characters "GEN" (for example, GENMODEL is a general module used to compute the probability of finding a server busy, using Erlang C formula (m/m/c), and can be used by any component of CPExpert).

There are two special members (titled **AAAAAAAA** and **ZZZZZZZZ**) located in SOURCE. These two members describe the most recent release level (and PTF, if appropriate) of CPExpert code contained in SOURCE.

Finally, there are two modules that contain a log of changes to the CPExpert code: the CHANGES module and the CHANGESS[2] module. The CHANGES module lists the changes with the current release of CPExpert (except when the change is minor). Interested users can review the CHANGES log to see what code changes have been made with the current release, and can easily identify new guidance variables. The CHANGESS module is an accumulation of changes for multiple releases of CPExpert.

[2]This approach simply follows the approach established with MXG. The CHANGES and CHANGESS modules were added with CPExpert Release 13.2.

Chapter 2: Allocate space for CPExpert data sets.

The second step is to allocate space for SAS libraries maintained by CPExpert. CPExpert maintains SAS data sets to contain "templates" of the rules that might be produced during analysis, to describe problems that are discovered, and to contain historical information. These data sets contain the results produced by each CPExpert component processing system measurement data.

Exhibit 2-2 illustrates the JCL required to allocate space for CPExpert SAS libraries. SAS will automatically generate optimal DCB parameters.

```
//jobname     JOB     job card information
//STEP01      EXEC    IEFBR14
//CPEDATA     DD      DSN=prefix.CPEXPERT.CPEDATA,DISP=(,CATLG),
//                    VOL=SER=xxxxxx,UNIT=3380,SPACE=(CYL,(10,1))
//HISTORY     DD      DSN=prefix.CPEXPERT.HISTORY,DISP=(,CATLG),
//                    VOL=SER=xxxxxx,UNIT=3380,SPACE=(CYL,(5,1))
```

JOB CONTROL LANGUAGE TO ALLOCATE DASD SPACE

EXHIBIT 2-2

